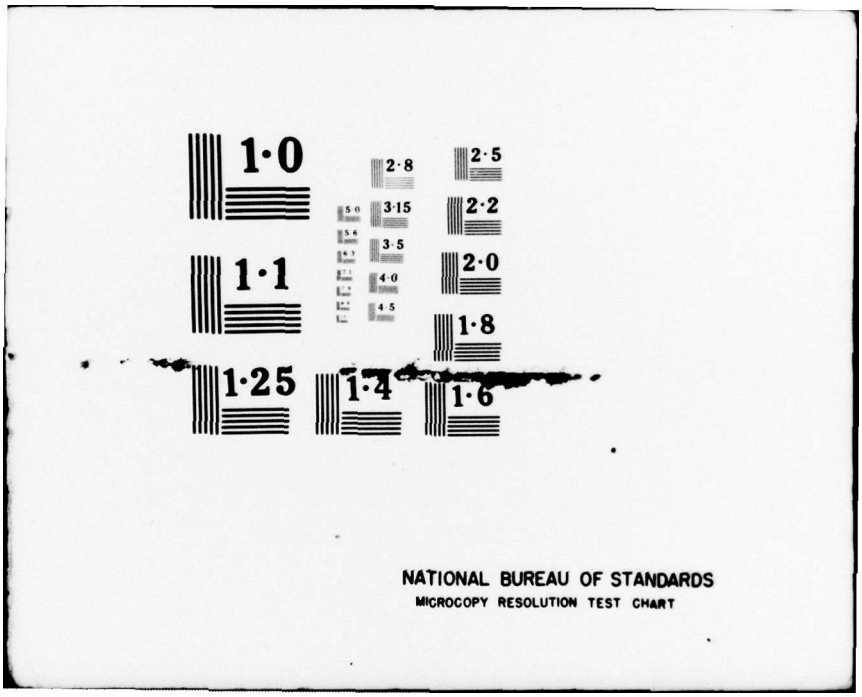
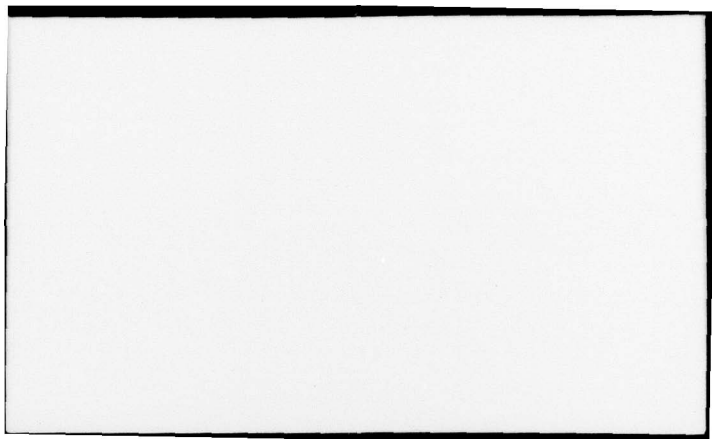
AD-A046 404 WHARTON SCHOOL OF FINANCE AND COMMERCE PHILADELPHIA P--ETC F/G 12/2
DESCRIPTION OF THE WHARTON/ODA SYSTEM.(U)
NOV 77 N00014-75-C-0440
UNCLASSIFIED 77-11-02 NL

| OF |
ADA
046404

END
DATE FILMED
12-77
DDC


1·0
1·1
1·25
1·4
1·6
1·8
2·0
2·2
2·5
2·8
3·15
3·5
4·0
4·5
5·0
5·6
6·3
NATIONAL BUREAU OF STANDARDS
MICROCOPY RESOLUTION TEST CHART



DESCRIPTION OF THE WHARTON/ODA SYSTEM

by

Operational Decision Aiding
Project Staff

E. G. Hurst, Jr., Editor

Working Paper 77-11-02

Department of Decision Sciences
The Wharton School
University of Pennsylvania

This description has been prepared under the Operational Decision Aiding Project, supported by ONR Contract No. N00014-75-C-0440. Any comments and suggestions would be welcome.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER (14) 77-11-02	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle) (6) DESCRIPTION OF THE WHARTON/ODA SYSTEM,	5. TYPE OF REPORT & PERIOD COVERED (9) Final rept.
6. PERFORMING ORG. REPORT NUMBER 77-11-02	
7. AUTHOR(s) Operational Decision Aiding Project Staff	8. CONTRACT OR GRANT NUMBER(s) (15) N0014-75-C-0440
9. PERFORMING ORGANIZATION NAME AND ADDRESS Decision Sciences Department University of PA/Wharton School Philadelphia, PA 19104	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS Technical report
11. CONTROLLING OFFICE NAME AND ADDRESS Office of Naval Research Information Systems Arlington, Virginia 22217	12. REPORT DATE (11) Nov 77 11/77
	13. NUMBER OF PAGES (12) 11p.
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report) Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Unlimited

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Decision Aiding
Computer Systems
Windows
Communication protocols

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The purpose of this document is to give a brief introduction to the hardware available in the Wharton Computer Center and the Decision Aiding Systems Laboratory (DASL), and to summarize the software which has been developed at Wharton under the Operational Decision Aiding Project. More detail about most of these features is available; in particular, some of the other software developed for use on the Wharton system may be of interest to other contractors, now that the system is easily accessible via the ARPANET.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

408 757

DESCRIPTION OF THE WHARTON/ODA SYSTEM

Introduction

The purpose of this document is to give a brief introduction to the hardware available in the Wharton Computer Center and the Decision Aiding Systems Laboratory (DASL), and to summarize the software which has been developed at Wharton under the Operational Decision Aiding Project. More detail about most of these features is available; in particular, some of the other software developed for use on the Wharton system may be of interest to other contractors, now that the system is easily accessible via the ARPANET. Inquiries should be addressed to Gerry Hurst or any other member or the Wharton project team.

Hardware and Operating System

DECsystem-10

The Wharton DECsystem-10 consists of a KI10 processor made by Digital Equipment Corporation, 192K words of core storage, 410 megabytes of online disk storage on 3 disk drives, a 9-track 800BPI tape drive, two line scanners which handle 128 terminal lines, a 300 lines-per-minute printer, and an IMP-10 and Pluribus VDA adapter which connect to the ARPANET. Version 6.03 of the TOPS-10 operating system is currently being run on the DEC-10, and includes the following features:

Personalized login and control -- Individual users are recognized separately from the one or more project numbers to which they have access. This feature permits personalization of mail and system notices, program use history, budgetary control, and the like.

Interface to the ARPANET -- Any program can use the ARPANET as a standard DEC-10 I/O device. Users logged in on the DEC-10 can log in at remote hosts using the Telnet protocol. Remote users can similarly login in on the DEC-10 through the ARPANET. It is also possible to send mail, and transfer files to and from remote hosts.

Virtual Memory -- Once a program has exceeded the physical capacity for storage on the DEC-10, it is possible for it to use "virtual memory", which increases the storage capabilities to the 18-bit address space of the DEC-10. This feature also allows easier control of programs which would normally take nearly all of physical memory to run. These programs would hog the valuable resource of core memory, but by forcing a program to go virtual before reaching the physical limits of the machine, all users see improved response.

Large assortment of languages and special programs -- available on the Wharton DEC-10 are: APL, BLISS, COBOL, FORTRAN, LISP, MACRO-10, and POP. Canned program packages include PLANETS and SPSS. A large library of FORTRAN and APL functions and programs is available. Finally, a wide assortment of unique experimental software for planning, office automation, data base management, and so forth are in various stages of development.

Computer Terminal Devices

The terminal devices available in the DASL include an ADDS MRD400 terminal, a Datamedia 1520A terminal, and a Grinnell GMR-26 color raster graphics display.

The Datamedia 1520A is a standard, white on black CRT terminal. The screen is 24 x 80 charaters.

The ADDS MRD400 is an eight color display generator with a full alphanumeric character set. The ADDS uses a standard RGB color television monitor display. The screen is 24 x 80 characters. The ADDS also has a limited graphics capability, making it suitable for display of bar charts and trend curves. Software is available for formatting CRT screen displays and menu selection capablity.

The Grinnell is a 480 x 512 color raster graphics device. It provides a full graphics capability. The software permits the construction of displays using four sizes of alphanumeric characters, lines, dots, rectangles, circles, discs, and triangles, as well as the filling of polygons. Software is currently being developed to allow the drawing of figures with a track ball.

Also available at Wharton for the use of project team members are a number of GP100 CRT terminals, which contain a microprocessor programmed to permit such operations as scrolling sections of the screen, color reversal, and the like. In addition, there are several hard copy terminals of various types on which reports such as this one can be produced.

Other Hardware in the DASL

The VOTRAX voice synthesizer is capable of producing 64 different phonemes, at four inflection levels, with commands sent by the computer. It does not in any way accept voice input. Within the DAISY system, it is used to provide certain outputs which get the attention of the user, which may not be focused on the screen. In particular, it interacts with the alert system to

warn the user of the occurrence of conditions of interest.

Physically, the VOTRAX is connected in series with the modem connecting a terminal to the DAISY system. It examines the incoming stream of characters sent by the computer. When two ^V characters are seen, it takes the following characters and loads its buffer. It continues loading until it sees two ^W characters, when it "says" the syllables in its buffer. There are also programs which can translate normal English text into the appropriate codes, with about 98% accuracy.

The two ADVENT Videobeam projectors are used to provide the capability of group displays. The ADVENTs are commercial, 7-foot color televisions. They project onto a specially constructed reflective screen, and accept input either from a normal television tuner, from standard video, or from three separate red, green, and blue (RGB) video signals. Such signals are provided by the ADDS terminal and the Grinnell graphics system.

The trackball device is an integral part of the Grinnell graphics system. It permits the user to position a visible cursor precisely anywhere on the face of the graphics screen. With software which is already available for the RAMTEK, it can also be used for drawing continuous contours on the screen, and for selection from menus shown by the programs.

The joystick device, similar in function to the trackball, is connected to the standard terminals for control of the window displays, and for menu selection. Many of the functions now performed by written commands, such as EXPAND, SHRINK, and MOVE windows, will be more easily specified by using the joystick to position a cursor in the appropriate position on the screen, and then giving the command.

The DASL also has available facilities for videotaping any of the displays, or for playing back videotapes. The standard used is the U-matic cassette, a 3/4" color (or black and white) videotape with two audio tracks. There are several recording and playback units available, along with sophisticated editing equipment to permit contractors to videotape their demonstrations, mixing signals from the two terminals (graphics and character) and signals from color or black and white cameras. Most of the equipment is SONY, but it is compatible with that provided by other manufacturers such as JVC or Panasonic. (Note that Betamax is a 1/2" cassette and is <u>not</u> compatible.)

Software Developed for the ODA Project

Windows

The major disadvantage of interactive systems which operate through screen terminals is that the user is unable to control what remains on the screen and what does not. The window package allows the user to define and control windows, portions of the screen each of which behaves as a separate terminal. Thus information of transient interest may be directed to one window which is being frequently used, while more important text may be preserved in a relatively stable window. In addition to allowing this degree of control over window configurations, the window package may also be used to allow the user to interact with several asynchronous processes. For example, one window may be used to converse with a colleague who is currently working on the system, a second may be used to receive alerts and other useful news, while a third may be used to perform the task at hand, for example editing. Windows may be dynamically defined and moved as required by the user, or the user may relinquish control of certain window formatting tasks to a program. The window package contains three main components: The code to manipulate and define windows; a small operating system which allows for asynchronous processing; and a set of functions, the PTY package, which enables control of other jobs.

Window parameters -- The windows themselves are best described by listing the components of the data structure used to define a window:

1. TOPLX, TOPLY, SIZEX, SIZEY define the coordinates of the top left hand corner of the window and the lengths of its sides.

2. BORDER is a character string used to describe the border. If there is no border, the space is used for output.

3. TITLE describes the title (if any) for the window.

4. SCROLLP describes the ouput mode of the window: Clear and write from the top, scroll, or wrap around.

5. SCROLLN defines, in case the output mode is scroll, the number of lines of text which will be remembered for the window. (This will be larger than the number of lines currently available in the window)

6. READP, if set to 0, does not allow the user to input information to that window.

7. BLINKP blinks text in the window.

8. FORECOL and BACKCOL describe the foreground and background colors for color terminals.

Commonly used procedures include DISPLAY, CLEAR, EXPAND in addition to the functions which are used for I/O to each window. DAISY contains several simple window manipulating commands which may be examined through the HELP facility.

Window package --The window package is written in POP which is an incrementally compiled language. A basic procedure in the window package is ERRCOM, which compiles input from one window, sends output to a second, and error messages to a third. Through the input window, debugging of the DAISY code itself may be accomplished. The operating system is quite simple and enables various tasks to be run repeatedly at defined frequencies. It is possible to add and remove tasks as required.

PTY package -- The PTY package allows windows to be connected to pseudo teletypes, enabling the user to converse with several jobs at once. Data are transferred to and from the PTYs by the operating system, and the user may input to any of these windows. Output to each is asynchronous and discharged when it is available, though it may be suppressed when required. There are special procedures which allow a PTY to be logged in, connected to a window, silenced, logged out, etc. It is possible to send control characters to the PTY.

Database

One of the ONR contractors, CTEC, was tasked to provide a description of a suitable data set for experimental test of the decision aids in the context of the ONRODA scenario, and to provide instances of such data. The University of Pennsylvania has structured the data and stored them in a CODASYL network database.

The database currently contains information about operations areas (terrain, weather, ocean characteristics, etc.) and platform, sensor, weapons, and communication characteristics. The database is also set up to contain information about Naval Order of Battle Plans, logistical information, and status of forces. Such information can be added as required for the testing of the decision aids.

For further information on the database, see the CTEC final report on the ODA database, and a commented Data Definition Language file describing the database structure in

ONR.DDL[4010,200].

The database is maintained using the SEED database management system. Information about how to access the database, and manuals on SEED and an interactive data management language, are available from Wharton.

Communications

Because communication can be very general, it was necessary to develop communications protocols to ensure an orderly process of passing messages. These protocols are implemented in several languages to allow rapid incorporation of new modules. Three communication situations have been identified:

Job to Job -- Jobs of equal status communicate with each other via files, using a file-communication protocol. Standard front-end modules that take care of all the housekeeping details are available in several languages, presenting jobs with a very simple interface. The protocol allows jobs to pass messages which are usually character strings. Arbitrarily large numbers of messages may be queued in each file (there are two files, one for each direction, associated with each communication channel). Messages are written and read sequentially, and procedures are provided for finding out the state of the files, e.g. the existence of unread messages. A handshaking protocol ensures that files are correctly opened and that channels are not confused. Jobs may also log in and control other jobs via the PTY package; see the discussion above under windows.

Human to job -- A user may communicate directly with programs via a PTY to set up new jobs, to send queries to databases, alerting systems, and the like, to receive responses, and to receive alert messages. It is also possible for the user to save the status of DAISY, exit to run other programs, and return and resume from the point of departure. A number of commands are provided in DAISY to facilitate this process. For example, LOGIN allows the DAISY user to log in a job and communicate through a given window to that job. The window effectively becomes a separate terminal and it is then possible to have several logical terminals on one physical terminal. Another command is STARTUP, which not only logs in a job, but also causes various programs to be run. For example, if no alerter service is active, it may be started in this manner. Thereafter, communication may continue through the job-to-job methods described above. Of course, all the interactive capability normally available on the system is at the disposal of the user.

Human to human -- Users at different locations can communicate directly with each other via the SPEAK system, through jobs they are running, and also through the extensive mail service available on the system.

Alerters

Alerters are dynamically defined programs that monitor databases for the occurrence of user-defined conditions. Whenever such a condition becomes true the alerter reports this to the user program.

The earliest work on alerters in the DAISY project is in the LDEMON program, which provides a simple data management system together with facilities for defining alerters. This system is written in LISP. Recently, alerting has been added to the WAND database management system, which has the advantage of the more powerful database facilities of the CODASYL DBTG report provided to the user of alerters.

A Network Alerter Service (NAS) is currently being developed to permit alerting on remote databases accessible through the ARPANET. This will be tested initially on the Wharton DEC-10 using LDEMON, WAND with alerters, and a fixed format file alerting program. It will then be extended to other, remote, databases through the DATACOMPUTER on the ARPANET.

Data Dictionaries

For several reasons, among them the ability to perform complex alerting, methods for providing structured descriptions of databases and making these descriptions available to the user are being explored. Such a description has been used to implement a simple query language which allows the user to interrogate a WAND database. As well as providing simple primitives for data extraction, the language also answers questions about the structure and definition of the database. For example, it is possible to give the command "DESCRIBE SHIP," meaning display the types of available information which pertain directly to ships. In addition, one may say "DESCRIBE CTEC," which is a request for a description of the major entities in the CTEC database, and also "DESCRIBE JFK," which sends a database query to WAND to display the particular information associated with the ship JFK. Having established the context of a particular ship by means of a DESCRIBE command, it is then possible to say "SHOW PLANES" which is understood to be a request for all planes associated with (i.e., aboard) the JFK.

The basis for this program is:

1. A copy of the schema of the database which has attached to it common names and other comments which may be of help in understanding the schema.

2. A dictionary which relates these names to the appropriate entity in the database or database schema.

Database primitives which will allow the data dictionary to be integrated with, and become part of, the database itself are now being explored.

Decision Structures

Related to this is the concept of decision structures, which provide a framework for exploring, and making decisions in, a data-rich environment. As well as incorporating a data dictionary, the system is intended to note what decisions have been made, what decisions are outstanding, and, in general, to keep track of the whole decision process. It is also required to enforce a set of constraints in this process which will correspond to those encountered in the environment being modelled. At present a set of programs which allow a user to explore, modify and create a decision tree have been implemented. It is hoped that some of the notions described above about database primitives can be incorporated.